5COSC011C - Tutorial 6-7 Exercises

As part of this tutorial for this week, you should complete **ALL** the tasks described in the following links and specifications:

1. **AsyncTask**
2. **IBM Watson Language Translator**
   In this tutorial exercise, you will learn how to use the Language Translator provided by IBM. Make sure that you understand all the code.

   a) Create a free IBM Cloud account at the following website: https://cloud.ibm.com/
   b) After you login, click on the "IBM Cloud" link on the top left and from the navigational menu choose **Dashboard**.

      Add the following 2 Web Services (Resources) to your account: "Language Translator" and "Text to Speech" (choose the Free Service for both).

   c) Modify the manifest file to include the Internet permission:

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="uk.ac.westminster.cs.ibmwatsonlanguagetranslateexample">
    <uses-permission android:name="android.permission.INTERNET"/>

    <application
    ....
```

   d) Add a `TextView` in your layout of your activity:

```
...
<TextView
    android:id="@+id/tv"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Hello World!"
...
```

e) Modify the `build.gradle` file (`Module: app`) to include the following:

```
implementation 'com.ibm.watson.developer_cloud:android-sdk:0.5.2'
implementation ('com.ibm.watson:ibm-watson:8.2.1') {
    exclude group: 'com.google.code.findbugs', module: 'annotations'
 }
```

f) Your activity should include:

```
import androidx.appcompat.app.AppCompatActivity;

import android.os.AsyncTask;
import android.os.Bundle;
import android.widget.TextView;

import com.ibm.cloud.sdk.core.security.Authenticator;
import com.ibm.cloud.sdk.core.security.IamAuthenticator;
import com.ibm.watson.language_translator.v3.LanguageTranslator;
import com.ibm.watson.language_translator.v3.model.TranslateOptions;
import com.ibm.watson.language_translator.v3.model.TranslationResult;
import com.ibm.watson.language_translator.v3.util.Language;

public class MainActivity extends AppCompatActivity {
    private LanguageTranslator translationService;
    TextView tv;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        tv = findViewById(R.id.tv);
        translationService = initLanguageTranslatorService();
        new TranslationTask().execute("Hello World and my friend");
    }


    private LanguageTranslator initLanguageTranslatorService() {
        Authenticator authenticator
                = new
IamAuthenticator(getString(R.string.language_translator_apikey));
        LanguageTranslator service = new LanguageTranslator("2018-05-
01", authenticator);

service.setServiceUrl(getString(R.string.language_translator_url));
        return service;
    }
```

```
    private class TranslationTask extends AsyncTask<String, Void,
String> {
        @Override
        protected String doInBackground(String... params) {
            TranslateOptions translateOptions = new
TranslateOptions.Builder()
                    .addText(params[0])
                    .source(Language.ENGLISH)
                    .target("es")
                    .build();
            TranslationResult result
                    =
translationService.translate(translateOptions).execute().getResult();
            String firstTranslation =
result.getTranslations().get(0).getTranslation();
            return firstTranslation;
        }

        @Override
        protected void onPostExecute(String s) {
            super.onPostExecute(s);
            tv.setText(s);
        }
    }
}
```

In the above code, make sure that you replace `R.string.language_translator_apikey` **with your personal API key for the web service (you can find this in your personal account in IBM Cloud under "Dashboard"->"Services"->"Language Translator" link.**

Also replace `R.string.language_translator_url`**, with your personal URL found in the same area of the Dashboard**.

The full documentation for the IBM Language Translator web service can be found in:
https://cloud.ibm.com/docs/language-translator?topic=language-translator-gettingstarted

And the API documentation in:
https://watson-developer-cloud.github.io/java-sdk/docs/master/index.html?com/ibm/

3. **IBM Watson Text To Speech**
   In this tutorial exercise, you will learn how to use the Text To Speech service provided by IBM. Make sure that you understand all the code.

   a) Login to the IBM Cloud website. Click on the "IBM Cloud" link on the top left and from the navigational menu choose **Dashboard**.

Add the following Web Service (Resource) to your account, if you did not do this during the previous tutorial exercise the language translator: "Text to Speech" (choose the Free Service).

b) Modify the manifest file to include the Internet permission:

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="uk.ac.westminster.cs.ibmwatsontexttospeechexample">
    <uses-permission android:name="android.permission.INTERNET"/>

    <application
    ....
```

c) Add a `Button` in your layout of your activity:

```
...
<Button
    android:id="@+id/bt1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Speak"
    android:onClick="buttonClick"
...
```

d) Modify the `build.gradle` file (`Module: app`) to include the following:

```
implementation 'com.ibm.watson.developer_cloud:android-sdk:0.5.2'
implementation ('com.ibm.watson:ibm-watson:8.2.1') {
    exclude group: 'com.google.code.findbugs', module: 'annotations'
 }
```

e) Your activity should include:

```
package uk.ac.westminster.cs.ibmwatsontexttospeechexample;

import androidx.appcompat.app.AppCompatActivity;

import android.os.AsyncTask;
import android.os.Bundle;
import android.view.View;

import com.ibm.cloud.sdk.core.http.HttpMediaType;
import com.ibm.cloud.sdk.core.security.Authenticator;
import com.ibm.cloud.sdk.core.security.IamAuthenticator;
import
com.ibm.watson.developer_cloud.android.library.audio.StreamPlayer;
import com.ibm.watson.text_to_speech.v1.TextToSpeech;
import com.ibm.watson.text_to_speech.v1.model.SynthesizeOptions;
```

```
import java.util.Calendar;
import java.util.GregorianCalendar;


public class MainActivity extends AppCompatActivity {
    private StreamPlayer player = new StreamPlayer();
    private TextToSpeech textService;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        textService = initTextToSpeechService();
    }

    public void buttonClick(View view) {
        Calendar cal = new GregorianCalendar();
        int hours = cal.get(Calendar.HOUR_OF_DAY);
        int minutes = cal.get(Calendar.MINUTE);

        String prefix = "";
        if (minutes < 9 && minutes >= 1)
            prefix = " O ";
        String suffix = " am ";
        if (hours > 12) {
            hours = hours - 12;
            suffix = " pm ";
        }
        new SynthesisTask().execute("Hello world. Good morning my
friend. How is the weather today? The time is " +
                + hours + " " + prefix + minutes + suffix);
    }

    private TextToSpeech initTextToSpeechService() {
        Authenticator authenticator = new
IamAuthenticator(getString(R.string.text_speech_apikey));
        TextToSpeech service = new TextToSpeech(authenticator);
        service.setServiceUrl(getString(R.string.text_speech_url));
        return service;
    }

    private class SynthesisTask extends AsyncTask<String, Void,
String> {

        @Override
        protected String doInBackground(String... params) {
            SynthesizeOptions synthesizeOptions = new
SynthesizeOptions.Builder()
                    .text(params[0])
                    .voice(SynthesizeOptions.Voice.EN_US_LISAVOICE)
                    .accept(HttpMediaType.AUDIO_WAV)
```

```
                    .build();
player.playStream(textService.synthesize(synthesizeOptions).execute()
.getResult());
            return "Did synthesize";
        }
    }
}
```

In the above code, make sure that you replace `R.string.text_speech_apikey` **with your personal API key for the web service (you can find this in your personal account in IBM Cloud under "Dashboard"->"Services"->"Text to Speech" link.**

Also replace `R.string.text_speech_url`**, with your personal URL found in the same area of the Dashboard**.

The full documentation for the IBM Text To Speech web service can be found in:
https://cloud.ibm.com/docs/services/text-to-speech?topic=text-to-speech-gettingStarted#getting-started-tutorial

And the API documentation in:
https://watson-developer-cloud.github.io/java-sdk/docs/master/index.html?com/ibm/

4. **AsyncTask and AsyncTaskLoader**
5. **Connecting to the Internet**
   In this exercise you will be implementing the Internet connection example which retrieves books from the Google Book API Web Service and it was covered in the lecture.

   a) Modify the manifest file to include the Internet permission:

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="uk.ac.westminster.networkexample">
    <uses-permission android:name="android.permission.INTERNET"/>

    <application
    ....
```

   b) The layout file:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/tv"
```

```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="24sp"
        android:text="Hello World!"
         />
</LinearLayout>
```

c) The activity:

```
package uk.ac.westminster.networkexample;

import android.net.Uri;
import android.os.AsyncTask;

import android.os.Bundle;
import android.util.Log;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.io.Reader;
import java.io.UnsupportedEncodingException;
import java.lang.ref.WeakReference;
import java.net.HttpURLConnection;
import java.net.URL;

public class MainActivity extends AppCompatActivity {

//https://www.googleapis.com/books/v1/volumes?q=pride+prejudice&maxRe
sults=5&printType=books

    final String DEBUG_TAG = "** MAIN_ACTIVITY **";
    // Base URL for the Books API.
    final String BOOK_BASE_URL =
            "https://www.googleapis.com/books/v1/volumes?";

    // Parameter for the search string
    final String QUERY_PARAM = "q";
    // Parameter to limit search results.
    final String MAX_RESULTS = "maxResults";
    // Parameter to filter by print type
    final String PRINT_TYPE = "printType";

    private TextView mTextView;
```

```
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        // Build up the query URI, limiting results to 5 printed
books.
        Uri builtURI = Uri.parse(BOOK_BASE_URL).buildUpon()
                .appendQueryParameter(QUERY_PARAM, "pride+prejudice")
                .appendQueryParameter(MAX_RESULTS, "5")
                .appendQueryParameter(PRINT_TYPE, "books")
                .build();

        String myurl = builtURI.toString();
        mTextView = findViewById(R.id.tv);
        mTextView.setText("Empty");

        new SimpleAsyncTask(mTextView).execute(myurl);
    }


    private String downloadUrl(String myurl) {
        InputStream inputStream = null;
        // Only display the first 50000 characters of the retrieved
        // web page content.
        int len = 500000;

        HttpURLConnection conn = null;
        try {
            URL url = new URL(myurl);
            conn =
                    (HttpURLConnection) url.openConnection();
            conn.setReadTimeout(10000 /* milliseconds */);
            conn.setConnectTimeout(15000 /* milliseconds */);
            // Start the query
            conn.connect();
            int response = conn.getResponseCode();
            Log.d(DEBUG_TAG, "The response is: " + response);
            inputStream = conn.getInputStream();

            // Convert the InputStream into a string
            String contentAsString =
                    convertInputToString(inputStream, len);
            return contentAsString;

            // Close the InputStream and connection
        } catch (IOException ex) {
            ex.printStackTrace();
        } finally {
            conn.disconnect();
            if (inputStream != null) {
                try {
                    inputStream.close();
                } catch (IOException ex2) {
```

```
                    ex2.printStackTrace();
                }
            }
        }

        return "";
    }



    // Reads an InputStream and converts it to a String.
    public String convertInputToString(InputStream stream, int len)
            throws IOException, UnsupportedEncodingException {
        StringBuilder builder = new StringBuilder();
        BufferedReader reader = new BufferedReader(new
InputStreamReader(stream));

        String line;
        while ((line = reader.readLine()) != null) {
            builder.append(line + "\n");
        }

        if (builder.length() == 0) {
            return null;
        }

        String resultString = builder.toString();
        return resultString;
    }


    public class SimpleAsyncTask extends AsyncTask<String, Void,
String> {
        private WeakReference<TextView> mTextView;

        SimpleAsyncTask(TextView tv) {
            mTextView = new WeakReference<>(tv);
        }

        @Override
        protected String doInBackground(String... strings) {
            String jsonFullString = downloadUrl(strings[0]);

            try {
                JSONObject data = new JSONObject(jsonFullString);
                JSONArray all_items = data.getJSONArray("items");
                // extract all items titles
                String titles = "";
                for (int i=0; i < all_items.length(); i++) {
                    JSONObject book = all_items.getJSONObject(i);
                    JSONObject volumeInfo =
book.getJSONObject("volumeInfo");
                    titles += i + ": " +
volumeInfo.getString("title") + "\n";
                }
```

```
                return titles;

            } catch (JSONException jex) {
                jex.printStackTrace();
            }

            return "";
        }


        protected void onPostExecute(String result) {
            mTextView.get().setText(result);
        }
    }
}
```

6. **AsyncTask Example with a Progress Bar**

In this exercise, you will be implementing the simulation of a long (time consuming) task with a progress bar. The example was covered during the lecture.

The layout file:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity"
    android:gravity="center">


    <TextView
        android:id="@+id/tv"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="AsyncTask Example"
        android:textSize="24sp"
         />
    <Button
        android:id="@+id/bt1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="40dp"
        android:text="Press to Start Background Task"
        android:onClick="buttonOClickHandler"/>

    <ProgressBar
        android:id="@+id/pb"
```

```
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="40dp"
        android:visibility="invisible"
        style="@android:style/Widget.ProgressBar.Horizontal"/>
</LinearLayout>
```

The code of the main activity:

```
package uk.ac.westminster.cs.asynctaskexample;

import androidx.appcompat.app.AppCompatActivity;

import android.os.AsyncTask;
import android.os.Bundle;
import android.view.View;
import android.widget.ProgressBar;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {
    ProgressBar pb;
    TextView tv;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        pb = findViewById(R.id.pb);
        tv = findViewById(R.id.tv);
    }

    public void buttonOClickHandler(View view) {
        tv.setText("Time consuming task in progress!");
        pb.setProgress(0);
        pb.setVisibility(View.VISIBLE);
        new MyAsyncTask().execute("Message sent to background
thread");
    }


    private class MyAsyncTask extends AsyncTask<String, Integer,
String> {

        @Override
        protected String doInBackground(String... strings) {
            for (int i=1; i<=5; i++) {
                try {

                    Thread.sleep(2000);
                    // notify user of progress in the UI thread
                    publishProgress(i);
                }
```

```
                catch (InterruptedException ex) {
                    ex.printStackTrace();
                }
            }

            return "Task completed";
        }

        /* Runs on the UI thread receiving the value sent by
publishProgress()
           run in the background thread */
        @Override
        protected void onProgressUpdate(Integer... values) {
            super.onProgressUpdate(values);
            pb.setProgress(values[0]*20);
        }


        @Override
        protected void onPostExecute(String s) {
            super.onPostExecute(s);
            pb.setVisibility(View.INVISIBLE);

            // Change text to whatever value returned from
doInBackground()
            tv.setText(s);
        }
    }
}
```